

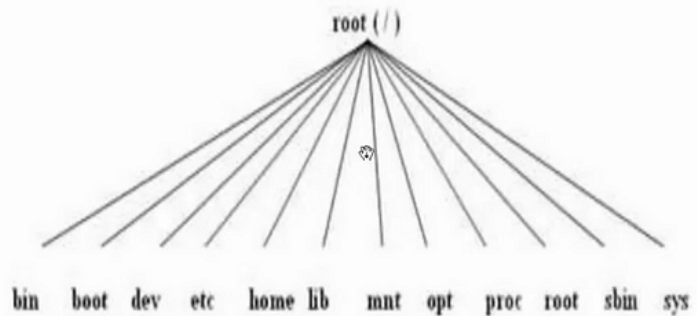


Photo by [Gabriel Heinzer](#) on [Unsplash](#)

## Linux File System Hierarchy

# FILESYSTEM HIRERACHY OF LINUX

- ❖ /bin Essential command binaries
- ❖ /boot Static files of the boot loader
- ❖ /dev Device files
- ❖ /etc Host-specific system configuration
- ❖ /lib for libraries and kernel modules
- ❖ /media Mount point for removable media
- ❖ /mnt Mount point for mounting temporarily
- ❖ /opt Add-on application software packages
- ❖ /sbin Essential system binaries
- ❖ /srv Data for services provided by this system
- ❖ /tmp Temporary files



## 1. /

The base of Linux Directory is the root. Every directory arises from root user. It is presented by forward slash(/)

```
[cloudshell-user@ip-10-4-14-241 ~]$ ls
aws  bin  boot  dev  etc  home  lib  lib64  local  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
```

## 2. /root

This is the home directory for root user

## 3. /bin

This contains binary executables. Common linux commands are stored in this folder

```
cloudshell-user@ip-10-4-14-241 bin$ ls
[
a2p          dbus-test-tool      gpgv2             machinectl        pod2text          shuf             trust
alias        dbus-update-activation-environment  gpg-zip           make              pod2usage         signer          tset
amazon-linux-extras  dbus-uuidgen        gpic              makedb            post-grohtml      skill            tsort
apropos      db_verify           grep              man               pr                slabtop         tty
arch         dd                 groff             mandb             preconv           sleep           tzselect
aria_chk     df                 groups            manpath           pre-grohtml       slogin          udevadm
aria_dump_log dgawk              grotty            mcookie           printenv          snice           ul
aria_ftdump  diff              groups            md5sum            printf            soelim          umask
aria_pack    diff              gsettings         mkdir             prlimit           sort            unmount
aria_read_log diff3              gsoelim           mkfifo            ps                sotrus          unalias
aria_read_log dir                gtar              mknod             psd               splain          uname
```

## 4. /sbin

This contains system binaries. The commands stored in this folder is used by System administrator for maintenance purpose.

```
[cloudshell-user@ip-10-4-14-241 ~]$ cd sbn
[cloudshell-user@ip-10-4-14-241 sbn]$ ls
access2      build-locale-archive  ctrlaltdel  fstpm      iconvconfig.x86_64  mkswap      piconv      setpci      unix_chkpwd
addnugphome  capsh               delpart     getcap     ifenslave          modinfo     phoenix_history_helper  sfdisk      unix_update
addpart      cfdisk              demod       getpcaps   init               modprobe    phoenixconv  shutdown    update-alternatives
adduser      chcpu               dmfilemapd  glibc_post_upgrade.x86_64  inssmd      newusers    rdisc        sln         useradd
agetty       chkconfig            dmsetup     iptables   install-info       nologin     readprofile  sudo        userdel
alternatives  cloudimg             dmccach     isettach   ldd                packer       reboot       swaplabel   usermod
amazon-linux-tools  chroot             faillock    groupmema  ldconfig            pam_console_apply  resizepart  swaponoff   vigr
applygnupg-defaults  clock              fdformat    groupmd5   losetup            pam_tally2     rmmod        swapon      vipw
```

## 5. /dev

This folder contains hardware Device files. It stores the information of usb, device attached to the system.

```
[cloudshell-user@ip-10-4-14-241 ~]$ cd dev
[cloudshell-user@ip-10-4-14-241 dev]$ ls
autofs          dm-13  dm-24  dm-35  dm-46  dm-6
btrfs-control   dm-14  dm-25  dm-36  dm-47  dm-7
bus             dm-15  dm-26  dm-37  dm-48  dm-8
cassandra       dm-16  dm-27  dm-38  dm-49  dm-9
```

## 6. /var

This contains variable files. It stores files which tend to grow such as log files.

/var/log = System log files generated by OS

/var/lib = contains database and package files

/var/mail = contains emails

/var/tmp = temporary files

## 7. /mnt

This directory is used to mount a file system temporarily

## 8. /media

Removable media devices

## 9. /usr

User binaries. Contains files and applications used by users.

```
[cloudshell-user@ip-10-4-14-241 usr]$ ls
bin  etc  games  include  lib  lib64  libexec  local  sbin  share  src  tmp
```

## 10. /etc

Configuration files. It contains configuration files of servers.

```
[cloudshell-user@ip-10-4-14-241 /]$ cd etc
[cloudshell-user@ip-10-4-14-241 etc]$ ls
adjtime      depmod.d      groff          init.d          logrotate.d    openldap        profile.d      rpc
aliases      DIR_COLORS    group          inputrc         machine-id     opt             protocols     rpm
alternatives  DIR_COLORS.256color  group-         issue           man_db.conf    os-release     python         rsyslog.d
bash_completion.d  DIR_COLORS.lightbgcolor  gshadow       issue.net       modprobe.d     pam.d           rc0.d         sasl2
bashrc        dracut.conf.d  gshadow-      krb5.conf       modules-load.d  passwd         rc1.d         securityty
binfmt.d      environment    gss           krb5.conf.d     motd           passwd-        rc2.d         security
chkconfig.d   ethertypes     host.conf     ld.so.cache     mtab           pkcs11         rc3.d         selinux
```

## 11. /boot

This directory contains files needed to boot the system

## 12. /home

Home directory, this contains secondary users home directory

## 13. /tmp

It contains temporary files created by System and users.

```
[cloudshell-user@ip-10-4-14-241 ~]$ cd tmp
[cloudshell-user@ip-10-4-14-241 tmp]$ ls
tmux-1000  v8-compile-cache-0
```

We will practice various commands:

### 1. touch: Touch to create a new file

<touch file.txt>

```
[cloudshell-user@ip-10-4-14-241 ~]$ touch file.txt
[cloudshell-user@ip-10-4-14-241 ~]$ ls
file.txt
```

### 2. create multiple files

touch python java reactjs

```
[cloudshell-user@ip-10-4-14-241 ~]$ touch python java reactjs
[cloudshell-user@ip-10-4-14-241 ~]$ ls
file.txt  java  python  reactjs
[cloudshell-user@ip-10-4-14-241 ~]$
```

### 3. create number of files

touch file{1..5}

```
[cloudshell-user@ip-10-4-14-241 ~]$ touch file{1..5}
[cloudshell-user@ip-10-4-14-241 ~]$ ls
file1  file2  file3  file4  file5  file.txt  java  python  reactjs
[cloudshell-user@ip-10-4-14-241 ~]$
```

### 4. For copy and paste

cp : to copy and paste

syntax: cp <option><source><destination>

- -r for recursive
- -v for verbose
- -f for forcefully

**cp file1 folder1**

```
[cloudshell-user@ip-10-4-14-241 ~]$ mkdir folder1
[cloudshell-user@ip-10-4-14-241 ~]$ cp file1 folder1
[cloudshell-user@ip-10-4-14-241 ~]$ cd folder1/
[cloudshell-user@ip-10-4-14-241 folder1]$ ls
file1
```

5. To remove file and directory

**rm -rvf /maharashtra/mumbai**

6. To move file or rename a file or directory

**mv test testers**

## User Management

1. Create user account

**useradd Asma**

```
[cloudshell-user@ip-10-4-14-241 ~]$ sudo useradd Asma
```

```
dbus:x:81:81:system message bus:/:/sbin/nologin
cloudshell-user:x:1000:997:~/home/cloudshell-user:/bin/bash
Asma:x:1001:1001:~/home/Asma:/bin/bash
[cloudshell-user@ip-10-4-14-241 ~]$
```

2. For switching user account

**su Asma**

3. For deleting user account

**userdel Asma**

## Group Management

A group is a collection of user account.

1. To add group

**groupadd Developers**

2. To add single member in the group

**gpasswd -a Asma Developers**

```
[cloudshell-user@ip-10-4-181-49 etc]$ sudo gpasswd -a Asma Developers
Adding user Asma to group Developers
```

3. For adding multiple members in a group

**sudo gpasswd -M John,Martha,Simba Developers**

## Linux File System Permission



There are three kinds of permission

1. Basic permission
2. Special permission
3. Access control list permission (ACL)

**ls -l**

```
[cloudshell-user@ip-10-4-181-49 folder1]$ ls -l
total 0
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Oct 31 18:26 file1
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Oct 31 18:26 file2
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Oct 31 18:26 file3
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Oct 31 18:26 file4
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Oct 31 18:26 file5
```

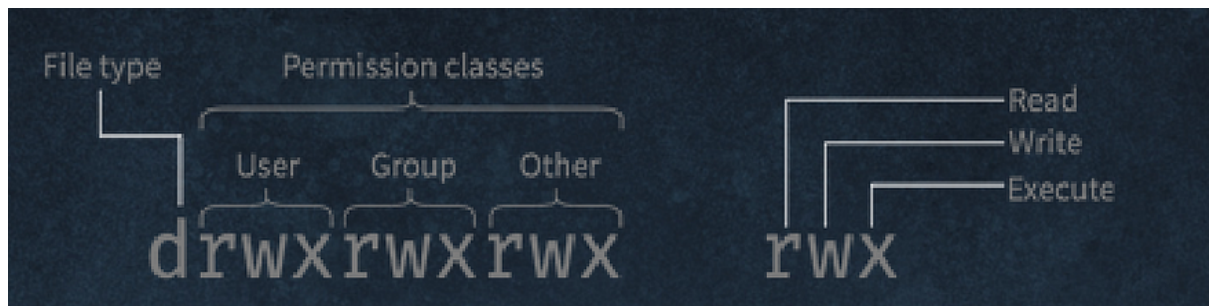
rw-rw-r — 1 cloudshell-user cloudshell-user 0 Oct 31 18:26 file1

- permission
- link
- owner
- group owner
- size of file
- date and time of file creation
- name of file

## Permission group

Permission description

- Owner(u) — Permissions of Owner
- Group(g) — Permissions for the members of Group
- Other(o) — Permission used by all Other users



Set permissions with numeric value

read(r) = 4

write(w) = 2

execute(x) = 1

Read(4)	Write(2)	Execute(1)	Number
R	W	X	7
R	W	-	6
R	-	X	5
R	-	-	4
-	W	X	3
-	W	-	2
-	-	X	1
-	-	-	0

For giving read write execute permission to owner and others for file2.txt

**sudo chmod 707 file2.txt**

```
[cloudshell-user@ip-10-4-189-76 folder1]$ sudo chmod 707 file2.txt
[cloudshell-user@ip-10-4-189-76 folder1]$ ls -l
total 0
-rw-rw-r-- 1 Hermoine      cloudshell-user 0 Nov  1 09:13 file1.txt
-rwx---rwx 1 cloudshell-user cloudshell-user 0 Nov  1 09:13 file2.txt
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Nov  1 09:13 file3.txt
```

For changing owner of file:

**sudo chown Hermoine file1.txt**

```
[cloudshell-user@ip-10-4-189-76 folder1]$ sudo chown Hermoine file1.txt
[cloudshell-user@ip-10-4-189-76 folder1]$ ls -l
total 0
-rw-rw-r-- 1 Hermoine      cloudshell-user 0 Nov  1 09:13 file1.txt
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Nov  1 09:13 file2.txt
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Nov  1 09:13 file3.txt
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Nov  1 09:13 file4.txt
-rw-rw-r-- 1 cloudshell-user cloudshell-user 0 Nov  1 09:13 file5.txt
[cloudshell-user@ip-10-4-189-76 folder1]$
```

## Access Control List:

It provides more flexible mechanism for file systems.

It is used for providing special permissions to users or groups.

To check ACL permission

### getfacl folder1

```
[cloudshell-user@ip-10-4-189-76 folder1]$ cd ..
[cloudshell-user@ip-10-4-189-76 ~]$ sudo getfacl folder1
# file: folder1
# owner: cloudshell-user
# group: cloudshell-user
user::rwx
group::rwx
other::r-x
```

To set ACL permission to user

### setfacl -s user:billy:r-x file2.txt

## Regular Expression:

Regular expressions are special characters which help searches data and match complex patterns

## grep:

(Global Regular Expression Print)- The grep filter searches a file for particular pattern of characters and display all lines that matches the pattern.

### grep -i delhi file1.txt

```
[cloudshell-user@ip-10-4-189-76 ~]$ vi file1.txt
[cloudshell-user@ip-10-4-189-76 ~]$ grep -i delhi file1.txt
Delhi
Delhi
Delhi
```

To search a string in a file

### grep root /etc/passwd

```
[cloudshell-user@ip-10-4-189-76 ~]$ grep root /etc/passwd
root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/sbin/nologin
```

To search a string in multiple files

### grep root /etc/passwd /etc/group

```
[cloudshell-user@ip-10-4-189-76 ~]$ grep root /etc/passwd /etc/group
/etc/passwd:root:x:0:0:root:/root:/bin/bash
/etc/passwd:operator:x:11:0:operator:/root:/sbin/nologin
/etc/group:root:x:0:
[cloudshell-user@ip-10-4-189-76 ~]$
```

To search a string in all files recursively

## **grep -v /etc/root**

### **Find command:**

It is one of the most important and used command in Linux. It is used to locate the list of files and directories based on the condition

To find files under home directory

### **find /home -name file2.txt**

```
[cloudshell-user@ip-10-4-189-76 ~]$ sudo find /home -name file2.txt
/home/cloudshell-user/folder1/file2.txt
[cloudshell-user@ip-10-4-189-76 ~]$
```

### **WC(Word Count):**

wc command is use to count the number of words and line numbers

### **wc -l /etc/passwd**

```
[cloudshell-user@ip-10-4-189-76 ~]$ wc -l /etc/passwd
18 /etc/passwd
[cloudshell-user@ip-10-4-189-76 ~]$
```

### **Head:**

It is used to find the top lines in a file

head /etc/passwd

```
[cloudshell-user@ip-10-4-189-76 ~]$ head /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
[cloudshell-user@ip-10-4-189-76 ~]$
```

head -n 15 /etc/passwd

```
operator:x:11:0:operator:/root:/sbin/nologin
[cloudshell-user@ip-10-4-189-76 ~]$ head -n 15 /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
```

Tail is used to find the last lines in a file.

**tail -n /etc/passwd**

```
[cloudshell-user@ip-10-4-189-76 ~]$ tail -n 5 /etc/passwd
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
cloudshell-user:x:1000:997:/:/home/cloudshell-user:/bin/bash
HarryPotter:x:1001:1001:/:/home/HarryPotter:/bin/bash
Hermoine:x:1002:1002:/:/home/Hermoine:/bin/bash
```

## Archiving Files in Linux:

Archiving is the process of combining multiple directories and files in a single file. It can be used as a form of backup.

### Tar:

tar stands for tape archive. It is used by system administrators

options for tar:

- -c for create
- -x for extract
- -v for verbose
- -f for forcefully
- -t for test
- -z for gzip

**tar <options><files>**

**tar -cvzf /mnt/backup.tar /var**



```
[cloudshell-user@ip-10-2-101-234 mnt]$ ls  
backup.tar
```

to show file size in human readable format

**du -sh /mnt/backup.tar**

```
[cloudshell-user@ip-10-2-101-234 ~]$ du -sh /mnt/backup.tar  
338M    /mnt/backup.tar
```

To extract a tar archive file in default location

**tar -xvf /mnt/backup.tar**

To create a tar archive file with compress in size(gzip)

**tar -cvf archive.tar.gz file1.txt file2.txt folder1**

```
[cloudshell-user@ip-10-2-101-234 ~]$ mkdir folder1 folder2  
[cloudshell-user@ip-10-2-101-234 ~]$ touch file{1..5}.txt  
[cloudshell-user@ip-10-2-101-234 ~]$ tar -cvf archive.tar.gz file1 file2 directory  
tar: file1: Cannot stat: No such file or directory  
tar: file2: Cannot stat: No such file or directory  
tar: directory: Cannot stat: No such file or directory  
tar: Exiting with failure status due to previous errors  
[cloudshell-user@ip-10-2-101-234 ~]$ tar -cvf archive.tar.gz file1.txt file2.txt folder1  
file1.txt  
file2.txt  
folder1/
```

To extract a tar archive file with compress in size gzip

## **Job Automation:**

Job automation in Linux refers to the process of automating repetitive tasks or jobs by creating scripts or using tools that can perform these tasks automatically without the need for manual intervention.

1. `at` command is used to execute one job at a time
2. `crontab` is used to execute a job multiple times

## **Managing networking :**

To show IP address : **`ipconfig`** or **`ip addr`**